H-1141

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent Application of

K. Serizawa et al.

ATTN: Manager,
Applications Branch

Serial No. 10/825,158

Filed: April 16, 2004

For: METHOD FOR ALLOCATING STORAGE AREA
TO VIRTUAL VOLUME

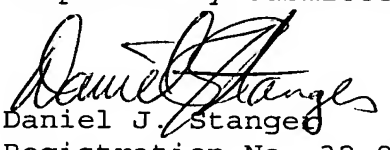
TRANSMITTAL OF CERTIFIED PRIORITY DOCUMENT

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

Submitted herewith is a certified priority document (JP 2003-150082) of a corresponding Japanese patent application for the purpose of claiming foreign priority under 35 U.S.C. § 119. An indication that this document has been safely received would be appreciated.

Respectfully submitted,



Daniel J. Stanger
Registration No. 32,846
Attorney for Applicants

MATTINGLY, STANGER & MALUR
1800 Diagonal Road, Suite 370
Alexandria, Virginia 22314
(703) 684-1120
Date: August 16, 2004

日本国特許庁
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されてる事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出願年月日
Date of Application: ２００３年　５月２８日

出願番号
Application Number: 特願２００３－１５００８２

[ST. 10/C]: [ＪＰ２００３－１５００８２]

願人
Applicant(s): 株式会社日立製作所

２００３年１１月２６日

特許庁長官
Commissioner,
Japan Patent Office

今井康夫

【書類名】　特許願
【整理番号】　K03001961A
【あて先】　特許庁長官殿
【国際特許分類】　G06F 12/00
【発明者】
【住所又は居所】　神奈川県川崎市麻生区王禅寺１０９９番地　株式会社日立製作所システム開発研究所内
【氏名】　芹沢　一
【発明者】
【住所又は居所】　神奈川県横浜市戸塚区戸塚町５０３０番地　株式会社日立製作所ソフトウエア事業部内
【氏名】　森田　眞司
【発明者】
【住所又は居所】　神奈川県川崎市麻生区王禅寺１０９９番地　株式会社日立製作所システム開発研究所内
【氏名】　岩見　直子
【特許出願人】
【識別番号】　000005108
【氏名又は名称】　株式会社　日立製作所
【代理人】
【識別番号】　100075096
【弁理士】
【氏名又は名称】　作田　康夫
【手数料の表示】
【予納台帳番号】　013088
【納付金額】　21,000円
【提出物件の目録】
【物件名】　明細書　1
【物件名】　図面　1

【物件名】　要約書　1

【プルーフの要否】　要

【書類名】　明細書

【発明の名称】　記憶領域割当方法、システム及び仮想化装置

【特許請求の範囲】

【請求項１】

計算機、前記計算機に接続される仮想化装置及び前記仮想化装置に接続される記憶装置において前記計算機に前記記憶装置が有する記憶領域を割当てる方法であって、

仮想化装置で予め前記計算機に仮想的な記憶領域を割当て、

前記計算機が前記仮想的な記憶領域を使用する際に、前記仮想的な記憶領域と前記記憶装置が有する記憶領域とを対応付けてその情報を保持し、前記情報を元に、前記計算機に割当てられた前記仮想的な記憶領域に前記記憶装置が有する記憶領域を割当てることを特徴とする記憶領域割当方法。

【請求項２】

請求項１記載の記憶領域割当方法であって、

前記仮想的な記憶領域に前記計算機がデータを書き込む時に、前記記憶装置が有する記憶領域から前記仮想的な記憶領域に使用されていない記憶領域を選択し、前記仮想的な記憶領域中の前記書込みの対象となる記憶領域に前記選択された記憶領域を割当てることを特徴とする記憶領域割当方法。

【請求項３】

請求項２記載の記憶領域割当方法であって、

前記仮想的な記憶領域に割当てられた前記記憶装置の有する記憶領域が所定の条件を満たす場合に、前記情報を更新し、前記仮想的な記憶領域に対する前記記憶装置の有する記憶領域の割当を止めることを特徴とする記憶領域割当方法。

【請求項４】

請求項３記載の記憶領域割当方法であって、

前記所定の条件とは、前記計算機が、前記仮想的な記憶領域に割当てられた前記記憶装置が有する記憶領域を使用しなくなったという条件であることを特徴とする記憶領域割当方法。

【請求項５】

請求項１記載の記憶領域割当方法であって、

前記計算機からの所定の書き込みである場合には、前記仮想的な記憶領域に割当てる前記記憶装置が有する記憶領域の容量を他の書き込みで割当てられる前記記憶装置が有する記憶領域の記憶容量と異ならせることを特徴とする記憶領域割当方法。

【請求項６】

請求項５記載の記憶領域割当方法であって、前記所定の書き込みとは、前記計算機によるファイルシステムの初期化処理の際に発生する書き込みであることを特徴とする記憶領域割当方法。

【請求項７】

請求項６記載の記憶領域割当方法であって、前記ファイルシステムがジャーナル領域を使用する場合に、前記ジャーナル領域へ前記計算機が書き込んだ情報に対応したファイルシステム上のファイルの消去または縮小を特定し、この消去または縮小によって、ファイルシステム上のどのデータも割当てられなくなった前記仮想的な記憶領域に対応する前記記憶装置が有する記憶領域を、前記仮想的な記憶領域への割当から外すことを特徴とする記憶領域割当方法。

【請求項８】

計算機及び記憶装置と接続される仮想化装置であって、

前記計算機と接続されるポートと、

前記記憶装置と接続されるポートと、

前記ポートから入力されるデータを他の前記ポートに転送する転送部と、

制御部と、

記憶部とを有し、

前記記憶部は、前記計算機に提供される仮想的な記憶領域と前記記憶装置が有する記憶領域との対応関係についての情報を保持し、

前記制御部は、前記ポートを介して前記計算機から前記仮想的な記憶領域に対する書き込み要求を受け付け、前記書き込み要求に対応する前記仮想的な記憶領域に前記記憶装置が有する記憶領域が割当てられているか否かを前記情報に基づいて判断し、前記仮想的な記憶領域に前記記憶装置が有する記憶領域が割当てら

れていない場合、未だ前記仮想的な記憶領域に割当てられていない前記記憶装置が有する記憶領域を選択し、前記仮想的な記憶領域に前記選択された前記記憶装置が有する記憶領域を割当て、前記情報の内容を更新することを特徴とする仮想化装置。

【請求項９】

請求項８記載の仮想化装置であって、前記制御部は、前記仮想的な記憶領域が前記計算機に使用されなくなった場合に、前記情報を更新して、使用されなくなった前記仮想的な記憶領域に割当てられていた前記記憶装置が有する記憶領域の割当を解消することを特徴とする仮想化装置。

【請求項１０】

請求項９記載の仮想化装置であって、前記制御部は、前記計算機からの書き込み要求の内容に応じて、前記仮想的な記憶領域に割当てる前記記憶装置が有する記憶領域の容量を決定することを特徴とする仮想化装置。

【請求項１１】

計算機及び記憶装置と接続される仮想化装置が有する記憶部に格納され、前記仮想化装置が有する制御部で実行されるプログラムであって、

予め前記計算機に仮想的な記憶領域を割当てるモジュールと、

前記計算機が前記仮想的な記憶領域を実際に使用する際に、前記仮想的な記憶領域と前記記憶装置が有する記憶領域とを対応付けてその情報を保持するモジュールと、

前記情報を元に、前記計算機に提供される前記仮想的な記憶領域に前記記憶装置が有する前記記憶領域を割当てるモジュールとを有することを特徴とするプログラム。

【請求項１２】

記憶装置と、

計算機と、

前記記憶装置及び前記計算機と接続される仮想化装置とを有し、

前記仮想化装置は、

予め前記計算機に仮想的な記憶領域を割当て、

前記計算機が前記仮想的な記憶領域を使用する際に、前記仮想的な記憶領域と前記記憶装置が有する記憶領域とを対応付けてその情報を保持し、前記情報を元に、前記計算機に提供される前記仮想的な記憶領域に前記記憶装置が有する記憶領域を割当てることを特徴とするシステム。

【請求項１３】

請求項１２記載のシステムであって、

更に前記仮想化装置に接続される管理用の計算機を有し、

前記仮想化装置は、前記管理用の計算機から入力される情報に基づいて前記仮想的な記憶領域を前記計算機に割当てることを特徴とするシステム。

【発明の詳細な説明】

【０００１】

【発明の属する技術分野】

本発明は、記憶装置システムが有する記憶領域の計算機への割当て技術、特に、仮想的な記憶装置システムにおける記憶領域の割当て技術に関する。

【０００２】

【従来の技術】

複数の記憶装置又は記憶装置システムを仮想化して一つの大規模な記憶装置システムとして計算機等に提供する装置が、特許文献１に開示されている。特許文献１によれば、装置において計算機等に提供される仮想的な記憶装置システムの仮想化の設定を変更することで、計算機に提供する仮想的な記憶領域のサイズを拡大することが出来る。

【０００３】

また、特許文献１における仮想的な記憶領域の拡大を計算機側で認識するためのソフトウエア技術に関しては、非特許文献１に開示されている。

【０００４】

【特許文献１】

特願２００２－３３０３２３号公報

【非特許文献１】

ＬＶＭ　ＨＯＷＴＯ（ｈｔｔｐ：／／ｗｗｗ．ｌｉｎｕｘ．ｏｒｇ／ｄｏｃｓ／

ｌｄｐ／ｈｏｗｔｏ／ＬＶＭ－ＨＯＷＴＯ－９．ｈｔｍｌから入手）の　９．９　Ｅｘｔｅｎｄｉｎｇ　ａ　ｌｏｇｉｃａｌ　ｖｏｌｕｍｅ

【０００５】

【発明が解決しようとする課題】

上述したように、特許文献１に示す仮想的な記憶領域のサイズを拡大する場合、非特許文献１に示すような計算機で使用されるソフトウェアが必須となる。これは、計算機が使用している記憶領域のサイズを動的に拡大することが可能である記憶装置又は記憶装置システムが従来存在せず、計算機がそのような記憶領域のサイズ拡大を想定していないためである。

【０００６】

しかし、このようなソフトウエアが必要になることは、以下のような管理コストの増大を引き起こす。即ち、計算機が複数ある場合に、その全てにソフトウエアを追加する必要があること、計算機の種類が複数ある場合それぞれに対応し相異なるソフトウエアを追加する必要があること、さらに、計算機で使用されるオペレーティングシステムがバージョンアップした場合に、それに追随して、対応するソフトウエアを更新する必要があるからである。

【０００７】

ところで、計算機が使用している記憶領域のサイズが拡大される理由は、最初から大きなサイズの記憶領域を計算機に割当てると、割当てられた記憶領域に未使用領域が発生して記憶領域に無駄が生じるので、計算機が使用するデータ量の増加に応じて記憶領域を割当てようとするからである。

【０００８】

したがって、計算機には最初から大きな記憶領域を割当てつつ、実際には割当てられた記憶領域の未使用領域を発生させないようにすれば、上述のような仮想的な記憶領域のサイズの拡大とそれに伴うソフトウエアの導入を行う必要がない。

【０００９】

つまり本発明の目的は、計算機に対して効率的に記憶領域を割当てることが可能な仮想化された記憶装置システム及び仮想化された記憶装置システムを計算機

に提供する装置を提供することである。

【００１０】

【課題を解決するための手段】

本発明は、計算機、計算機と接続され、仮想化された記憶領域を接続された計算機に提供する装置（以下「仮想化装置」）及び装置に接続される記憶装置とを有する構成とする。そして、仮想化装置は、計算機に提供する仮想化された記憶領域を作成する際に、記憶装置が有する記憶領域を実際には仮想化された記憶領域には割当てないが、計算機には割当てたように通知する。その後、計算機が仮想化された記憶領域へデータの書き込みを行ったときに初めて、仮想化された記憶領域中のデータの書き込みに対応する記憶領域に記憶装置の記憶領域を割当てる。

【００１１】

更に、仮想化装置は、仮想化された記憶領域に割当てられた記憶装置の記憶領域が計算機に使用されなくなった場合に、その記憶装置の記憶領域と仮想化された記憶領域との対応関係の情報を消去することでその記憶装置の記憶領域を仮想化された記憶領域から解放する。

【００１２】

尚、仮想化された記憶領域に割当てられる記憶装置の記憶領域の容量は、全て同じでも良く、割当てごとに異なる容量とする実施形態も考えられる。

【００１３】

【発明の実施の形態】

以下、本発明の実施形態を、図面を用いて説明する。

【００１４】

図１は、本発明を適用したシステムの第一の実施形態の全体構成を示す図である。

【００１５】

システムは、少なくとも１台のホストプロセッサ１２、少なくとも１台のストレージ装置１３、仮想化スイッチ１１及び管理コンソール１４を有する。

【００１６】

ホストプロセッサ１２は、ストレージ装置１３に格納されたデータを使用する計算機である。ホストプロセッサ１２は、仮想化スイッチ１１が提供する記憶領域を仮想化スイッチ１１に接続されていない他の計算機に提供する機能を有するファイルサーバでも良い。

【００１７】

ストレージ装置１３は記憶装置又は記憶装置システムである。ここで、記憶装置とは、ハードディスクドライブやDVDドライブ等の単体の記憶装置を指し、記憶装置システムとは、制御部及びハードディスクドライブ等のディスク装置を有するストレージサブシステムを指すものとする。また、ストレージ装置１３は、少なくとも１つの論理ユニット（以下「ＬＵ」）１３１を有する。ＬＵ１３１は、ストレージ装置１３が有する物理的な記憶領域から構成されている論理的な記憶領域である。ＬＵ１３１は、ホストプロセッサ１２などストレージ装置１３に接続される装置には、論理的に独立した１つのストレージ装置として認識される。

【００１８】

また、ＬＵ１３１は、複数の部分的な論理的記憶領域（以下「実領域」）１３２から構成される。実領域１３２の各々も、ストレージ装置１３が有する物理的な記憶領域に対応する。実領域１３２のサイズは任意であり、その範囲は、連続したアドレスをもつ領域である。

【００１９】

仮想化スイッチ１１は、図示するように通信線やスイッチによって他の装置と接続されており、他の装置と通信することができる装置である。また、仮想化スイッチ１１は、仮想化スイッチ１１自身に接続されている一つ又は複数のストレージ装置１３が有する記憶領域をまとめて一つまたは複数の記憶領域とする（以下「仮想化」と称する）仮想化装置である。そして、仮想化スイッチ１１は、仮想化した記憶領域を、仮想化スイッチ１１に接続されるホストプロセッサ１２等に提供する。

【００２０】

仮想化スイッチ１１とホストプロセッサ１２との間および仮想化スイッチ１１

とストレージ装置１３との間で使用される通信線やスイッチでは、ファイバチャネル等のプロトコルが使用される。ただし、使用される通信線やプロトコルはローカルエリアネットワーク等で使用される通信線やプロトコルでも良い。仮想化スイッチ１１は、ホストプロセッサ１２とストレージ装置１３との間に接続され、ホストプロセッサ１２が発行するコマンドをストレージ装置１３側に転送する機能を有する。また、仮想化スイッチ１１がホストプロセッサ１２に対して提供する仮想的な記憶領域を以下仮想ボリューム１００と称する。

【００２１】

仮想ボリューム１００は、少なくとも１つの実領域１３２から構成される仮想化された記憶領域である。仮想化スイッチ１１は、複数の仮想ボリューム１００をホストプロセッサ１２等に提供することが出来る。仮想ボリューム１００の各々には、仮想ボリュームを特定するための仮想化スイッチ１１内で一意の識別子（以下「仮想ボリューム識別子」）が与えられている。また、個々の仮想ボリューム１００の記憶領域には連続したアドレスが付されている。ホストプロセッサ１２は、ストレージ装置１３のＬＵ１３１内の実領域１３２を直接指定する代わりに、仮想ボリューム識別子及び仮想ボリューム１００内の場所を示すアドレスを指定して、ストレージ装置１３に格納されたデータを利用する。

【００２２】

管理コンソール１４は、仮想ボリューム１００を作成するためにシステム管理者によって使用される計算機であり、表示装置と入力装置を備える。管理コンソール１４は、仮想化スイッチ１１とネットワークを介して接続されている。

【００２３】

図２は、仮想化スイッチ１１の内部構成を示す図である。

【００２４】

仮想化スイッチ１１は、入力ポート２４０、出力ポート２５０、転送部２３０、制御部２１０及び記憶部２２０を有する。入力ポート２４０は、仮想化スイッチ１１がホストプロセッサ１２と通信するための通信線と接続されるポートである。出力ポート２５０は、仮想化スイッチ１１がストレージ装置１３と通信するための通信線と接続されるポートである。尚、入力ポート２４０及び出力ポート

２５０を構成する装置は同一であっても良い。この場合、どのポートを入力ポートあるいは出力ポートとして使用するかは、使用者が選択する。

【００２５】

転送部２３０はメモリを有し、そのメモリに転送情報テーブル２３１を保持する。転送情報テーブル２３１には、各入力ポート２４０を介して仮想化スイッチ１１と通信可能なホストプロセッサ１２及び各出力ポート２５０を介して仮想化スイッチ１１と通信可能なストレージ装置１３との間の対応関係についての情報が格納される。

【００２６】

転送部２３０は、転送情報テーブル２３１を参照し、入力ポート２４０がホストプロセッサ１２から受信した入出力要求を、要求先のストレージ装置１３と仮想化スイッチ１１との間の通信に使用される出力ポート２５０へ転送する。また、転送部２３０は、出力ポート２５０がストレージ装置１３から受信した応答情報やデータを、データ等を受信すべきホストプロセッサ１２と仮想化スイッチ１１の間の通信に使用される入力ポート２４０へ転送する。ただし、ホストプロセッサ１２から受け取った入出力要求が仮想ボリューム１００に対する入出力要求である場合には、転送部２３０は、後述のアクセス変換プログラム２１２に基づく制御部２１０の処理において選択されたストレージ装置１３へ入出力要求を送信する。

【００２７】

制御部２１０はプロセッサ及びメモリを有し、そのメモリに仮想ボリューム定義プログラム２１１、アクセス変換プログラム２１２及び割当処理プログラム２１３の各プログラムが格納される。これらのプログラムは、制御部２１０のプロセッサで実行される。

【００２８】

記憶部２２０は、仮想ボリューム管理テーブル２２１、実領域管理テーブル２２２及びアクセス変換テーブル２２４を記憶する。

【００２９】

アクセス変換テーブル２２４は、仮想化スイッチ１１がホストプロセッサ１２

に提供する仮想ボリューム１００ごとに存在する。アクセス変換テーブル２２４は、エントリ３３１と、対応する仮想ボリューム１００の仮想ボリューム識別子を登録するエントリ３３２とを保持する。個々のエントリ３３１には、仮想ボリューム１００内の記憶領域を示すアドレス範囲と、そのアドレス範囲に対応する実領域１３２が属するＬＵ１３１を指定する識別子であるＬＵアドレス及び実領域１３２のＬＵ１３１内における位置を示すＬＵ内アドレスとの対応関係の情報が登録される。即ちアクセス変換テーブル２２４は、仮想ボリューム１００の記憶領域のアドレスとストレージ装置１３の記憶領域のアドレスとの対応情報を保持している。

【００３０】

なお、仮想ボリューム１００に実領域１３２が割当てられていない場合、エントリ３３１のＬＵアドレス及びＬＵ内アドレスの情報が登録される部分には実領域１３２が割当てられていないことを示す情報、具体的には－1が登録される。アクセス変換テーブル２２４は、仮想ボリューム１００の記憶領域の構成が変更されたとき、制御部２１０によって更新される。

【００３１】

制御部２１０は、アクセス変換プログラム２１２を実行し、アクセス変換テーブル２２４を参照して、ホストプロセッサ１２から受け取った仮想ボリューム１００に対する入出力要求を、対応するストレージ装置１３が有するＬＵ１３１への入出力要求に変換する。また、対応するＬＵ１３１が存在しない場合、制御部２１０は、仮想ボリューム定義プログラム２１１を実行して、仮想ボリューム１００の定義変更処理を実行する。なお、性能向上のために、仮想化スイッチ１１は、アクセス変換テーブル２２４およびアクセス変換プログラム２１２を各入力ポート２４０ごとに有していても良い。

【００３２】

実領域管理テーブル２２２は、ＬＵ１３１ごとに存在する。実領域管理テーブル２２２は、ＬＵ１３１に含まれる実領域１３２を管理するために使用されるテーブルである。各実領域管理テーブル２２２には、ストレージ装置ＩＤ、ＬＵアドレス及び実領域リスト３２４が格納される。ストレージ装置ＩＤは、ＬＵ１３

１を保持するストレージ装置１３を示す識別子である。

【００３３】

実領域リスト３２４は、少なくとも１つのエントリ３２５を有する。各エントリ３２５は、ＬＵ１３１を構成する個々の実領域１３２に対応して設けられ、実領域ＩＤ、サイズ及び仮想ボリューム識別子の情報を登録する項目を有する。実領域ＩＤはエントリ３２５に対応する実領域１３２を特定するための識別子、サイズは実領域１３２のサイズである。またエントリ３２５に登録される仮想ボリューム識別子は、実領域１３２が割当てられている仮想ボリューム１００に割り振られた仮想ボリューム識別子である。実領域リスト３２４内のエントリ３２５は、実領域１３２のアドレス順に配列されている。

【００３４】

なお、本実施形態では、実領域１３２のサイズは固定であると仮定するため、エントリ３２５に実領域１３２のサイズについての情報を登録する項目を含めなくても良い。また、未使用である実領域１３２に対応するエントリ３２５の仮想ボリューム識別子を登録する項目には、未使用であることを示すnullが登録される。

【００３５】

このように、実領域管理テーブル２２２は、ＬＵ１３１に属する各実領域１３２が仮想ボリューム１００として利用されているか否かをに関する情報を保持しており、仮想化スイッチ１１が仮想化ボリューム１００に新たに割当てる実領域１３２を選択する際に使用される。

【００３６】

尚、実領域管理テーブル２２２は、仮想化スイッチ１１にストレージ装置１３が接続された場合等のタイミングで、管理端末１４を介した管理者の指示に基づいて作成される。さらに、この際に、ＬＵ１３１とそれを構成する実領域１３２の記憶容量等が決定される。尚、実領域管理テーブル２２２が作成された時点では、全てのエントリ２３５の領域ＩＤには仮想化スイッチ１１内で一意な識別子がそれぞれ書き込まれ、仮想ボリューム識別子にはnullが設定される。

【００３７】

仮想ボリューム管理テーブル２２１は、仮想ボリューム１００ごとに存在する。各仮想ボリューム管理テーブル２２１には、識別子エントリ３１１及び実領域リスト３１５が格納される。識別子エントリ３１１には、テーブル２２１に対応する仮想ボリューム１００の仮想ボリューム識別子が登録される。実領域リスト３１５は、テーブル２２１に対応する仮想ボリューム１００にどの実領域１３２が割当てられているかを示すリストである。実領域リスト３１５の各エントリ３１７は、仮想ボリューム１００上のアドレス順に並んでおり、そのアドレスに対応する実領域１３２の実領域ＩＤを格納している。当該仮想ボリューム１００のうち、実領域１３２が割当てられていない部分に相当する実領域リスト３１５のエントリ３１７には、有効な実領域ＩＤ３１７の代わりに空きを示すnull値が格納される。

【００３８】

このように、仮想ボリューム管理テーブル２２１は、仮想ボリューム１００の記憶領域がどの実領域１３２と対応付けられているかを示す情報を保持しており、仮想化スイッチ１１が解放可能な実領域１３２を選択するために使用される。

【００３９】

以下、本実施形態における仮想化スイッチ１１が行う記憶領域の割当て処理について説明する。

【００４０】

制御部２１０は、仮想ボリューム定義プログラム２１１を実行することで、仮想ボリューム１００の定義を作成または変更する。制御部２１０は、管理コンソール１４を介してシステム管理者から仮想ボリューム１００の作成要求を受け取り、仮想ボリューム管理テーブル２２１及びアクセス変換テーブル２２４を新規作成する。この場合、制御部２１０は、識別子エントリ３１１に、既に作成された別の仮想ボリューム１００と重複しない仮想ボリューム識別子を生成して格納し、実領域リスト３１５に空の値を設定することで実領域リスト３１５を初期化する。このように、仮想ボリューム１００の生成直後は、仮想ボリューム１００に実領域１３２が対応付けられない。従って、このとき制御部２１０は、新規作成される仮想ボリューム１００に対応するアクセス変換テーブル２２４にも空の

値を登録し、アクセス変換テーブル２２４を初期設定する。

【００４１】

このように初期設定することで、仮想ボリューム１００の作成時点では、ホストプロセッサ１２には所定の大きさの仮想ボリューム１００が割当てられたことを示す情報が通知されるが、その仮想ボリューム１００にはストレージ装置１３が提供する論理的な記憶領域である実領域１３２が割当てられていない状態となる。仮想化スイッチ１１は、その後、ホストプロセッサ１２等からのデータの書き込み要求を受信したタイミングで、仮想ボリューム１００に実領域１３２を割当てる。このようにすることで、計算機に割当てる記憶領域の無駄を省くことが出来る。

【００４２】

したがって、実領域１３２を仮想ボリューム１００に割当てるため、制御部２１０は、ホストプロセッサ１２のデータ書き込み要求等の要求に従い、仮想ボリューム管理テーブル２２１を変更する。この場合、制御部２１０は、仮想ボリューム１００に実領域１３２を割当てた後、アクセス変換テーブル２２４を更新する。尚、実際には、入力ポート２４０から受信した入出力要求の宛先が仮想ボリューム１００であった場合に、転送部２３０は、その入出力要求を制御部２１０に転送する。制御部２１０は、転送された入出力要求について、以下の処理を実行する。また、図３の処理終了後、制御部２１０は変換した入出力要求の要求先の情報を転送部２３０に送信し、転送部２３０はその情報に基づいて入出力要求を各ストレージ装置１３に転送する。

【００４３】

図３は、制御部２１０がホストプロセッサ１２から入出力要求を受取った際に行う処理の手順を示すフローチャートである。

【００４４】

まず、制御部２１０は、ホストプロセッサ１２から受け取った仮想ボリューム１００に対する入出力要求が書き込み要求であるかどうかを判断する（ステップ２００１）。

【００４５】

ホストプロセッサ１２からの入出力要求が書き込み要求である場合、制御部２１０は、書き込み要求で指定された仮想ボリューム１００のアドレスに実領域１３２が対応付けられているかどうかを、アクセス変換テーブル２２４で確認する（ステップ２００２）。指定された仮想ボリューム１００のアドレスに対応する実領域１３２がアクセス変換テーブル２２４に登録されていない場合、制御部２１０は、仮想ボリューム定義プログラム２１１を実行して仮想ボリューム１００の定義変更処理を行う。具体的には、制御部２１０は、指定された仮想ボリュームのアドレスに実領域１３２を割当てる。具体的には、実領域管理テーブル２２２から未使用の実領域１３２を検索して割当て（即ち実領域管理テーブル２２２および仮想ボリューム管理テーブル２２１を更新する）、アクセス変換テーブル２２４を更新する。尚、検索された実領域１３２の記憶容量が仮想ボリューム１００のアドレス領域に対して不足する場合、制御部２１０は更に空きの実領域１３２を検索し、充当するまで検索を行う（ステップ２００６）。

【００４６】

ステップ２００６の処理の後、ステップ２００２で実領域１３２が登録されていると判断された場合及びステップ２００１で書き込み要求ではないと判断された場合は、制御部２１０は、アクセス変換テーブル２２４を参照して、ホストプロセッサ１２から受け取った仮想ボリューム１００に対する入出力要求を対応するストレージ装置１３のＬＵ１３１への入出力要求に変換して（ステップ２００９）、処理を完了する。

【００４７】

このようにステップ２００６の処理によって、仮想ボリューム１００へデータが書き込まれる毎に、対応する実領域１３２を割当てることを可能にしている。

【００４８】

次に、本発明の第二の実施形態を説明する。

【００４９】

第一の実施形態の場合、実領域１３２が確保された後、その実領域１３２に格納されたデータが使用されなくなる場合について考慮されていない。そこで、第二の実施形態では、第一の実施形態に加え、使用されなくなった実領域１３２の

仮想ボリューム１００への割当てを止める（以下「解放する」）ことを考慮する。

【００５０】

本実施形態においては、第一の実施形態の構成に加え、制御部２１０のメモリには、デフラグ処理プログラム２１４が格納されている。制御部２１０は、システム管理者のデフラグ処理開始指示を管理コンソール１４を通して受け取り、デフラグ処理プログラム２１４を実行して、仮想ボリューム１００に格納されたファイル等のデータを再配置する。具体的には、制御部２１０は、ストレージ装置１３に格納されたファイルシステムの管理情報を読み取って、それに基づいて制御部２１０が実領域１３２のデータを空きの領域にコピーし、管理情報を書き換えることにより配置を変更する。なお、この再配置処理によるデータ破壊を防止するために、本実施形態ではシステム管理者は該仮想ボリューム１００を使用するファイルシステムを、デフラグ処理開始指示の前にアンマウントしておく必要がある。

【００５１】

その後、制御部２１０は、仮想ボリューム１００の記憶領域のうち、実領域１３２の割当てが不要になった記憶領域のアドレス範囲を特定し、特定した実領域１３２を解放し、仮想ボリューム管理テーブル２２１の対応するエントリ３１７を更新し、アクセス変換テーブル２２４の対応するエントリ３３１を更新する。

【００５２】

図４は、第二の実施形態における、実領域１３２を特定する処理の概念を示す図である。

【００５３】

図４は仮想ボリューム１００におけるファイルの配置及び仮想ボリューム１００と実領域１３２との対応関係を例示しており、図４（１）は制御部２１０のデフラグ処理プログラム２１４の実行前を、図４（２）はデフラグ処理プログラム２１４の実行後を示す。

【００５４】

図４（１）において、仮想ボリューム１００の図示された範囲の記憶領域には

３個のファイル５０１ａ、５０１ｂ及び５０１ｃが存在し、これらのファイルは、仮想ボリューム１００の記憶領域中、５０１ａ１、５０１ａ２、５０１ｂ及び５０１ｃの矩形の領域を占有している。一方、仮想ボリューム１００の図示された範囲の記憶領域には、２個の実領域１３２が対応付けられている。ここで、制御部２１０は、デフラグ処理プログラム２１４を実行して、各ファイルを仮想ボリューム１００におけるアドレス順に連続するように再配置する。

【００５５】

その結果、再配置処理後は、図４（２）のように、左（左の方がアドレスが小さいと仮定する）から順に５０１ａ１、５０１ａ２、５０１ｂ、及び５０１ｃに対応する記憶領域が連続して配置される。すると、５０１ｃの後ろにはまとまった空き領域が出現する（制御部２１０は各ファイルを再配置しているので、この空き領域の、仮想ボリューム１００上のアドレス範囲を再配置後のファイルシステムの管理情報から知ることが出来る）。この空き領域には実領域１３２を割当てておく必要がない。即ち、図４（２）で、実領域１３２－２は解放することが出来る。

【００５６】

制御部２１０は、デフラグ処理プログラム２１４の実行後、出現した空き領域の仮想ボリューム１００上のアドレス範囲から対応する実領域１３２を仮想ボリューム管理テーブル２２１から検索し、見つかった実領域１３２を解放する。具体的には、制御部２１０は該当する実領域１３２の情報を仮想ボリューム管理テーブル２２１から削除し、実領域管理テーブル２２２の該当する実領域１３２に対応する仮想ボリューム識別子をnullにし、アクセス変換テーブル２２４に登録された実領域１３２の情報を削除する。

【００５７】

次に、本発明の第三の実施形態を説明する。

【００５８】

第一の実施形態では、仮想ボリューム１００に割当てられる実領域１３２のサイズは固定であった。しかし、所定の処理、例えばフォーマット処理に基づくデータの書き込み要求では、比較的記憶容量の使用量が小さい処理が発生する。こ

の場合にも他の実施形態と同様に仮想ボリューム１００に固定サイズの実領域１３２を割当てると、割当てた記憶領域に無駄が発生する。そこで、本実施形態では、処理内容に見合ったサイズの実領域１３２を仮想ボリューム１００に割当てる。

【００５９】

本実施形態は第一の実施形態の構成に加え、制御部２１０のメモリにフォーマット処理プログラム２１５が格納されている。また、仮想ボリューム管理テーブル２２１のエントリ３１７に、対応する実領域１３２の記憶領域のサイズを示す情報が含まれる。また、実領域管理テーブル２２２のエントリ３２５に含まれる実領域１３２のサイズの情報は、本実施形態では第一の実施形態と異なり省略されない。エントリ３１７及びエントリ３２５にサイズの情報を含めるのは、上述したように、実領域１３２のサイズを可変長として、割当効率を上げるためである。

【００６０】

制御部２１０は、フォーマット処理プログラム２１５を実行することで、ホストプロセッサ１２に代わり、仮想ボリューム１００を使用するファイルシステムを初期化、即ちファイルシステム上のファイル及びディレクトリを全て消去し、新規にファイル及びディレクトリを作成可能な状態にする。このとき、仮想ボリューム１００には、メタデータと呼ばれる、管理領域が書き込まれる。このとき書き込まれるメタデータ１個のサイズはそれほど大きくないが、仮想ボリューム１００の記憶領域に一定の間隔に書き込まれるため、仮想ボリューム１００のサイズが大きい場合は多数のメタデータが書き込まれ、割当効率が低下する。実領域１３２のサイズを可変長とするのは、この割当効率の低下を防ぐためである。

【００６１】

尚、フォーマット処理プログラム２１５は、管理者等の管理端末１４を介した指示に基づき、あるいはホストプロセッサ１２からの指示に基づき実行される。

【００６２】

図５は、第三の実施形態において、ホストプロセッサ１２又は制御部２１０自身が生成した入出力要求を受け付けた際の制御部２１０の処理手順を示すフロー

チャートである。

【００６３】

本処理手順には、第一の実施形態の図３で示した処理手順に加え、制御部２１０が受信した入出力要求がフォーマット処理に基づく入出力要求かどうかを判断する処理（ステップ２００３）及びフォーマット処理である場合に、仮想ボリューム１００に割当てる実領域１３２のサイズを変更する処理（ステップ２００４）が含まれる。尚、その他のステップの処理は第一の実施形態と同様であるので、ここでは説明しない。

【００６４】

制御部２１０は、入出力要求で指定された仮想ボリューム１００のアドレスに対応する実領域１３２がアクセス変換テーブル２２４に登録されていない場合、その入出力要求がフォーマット処理に基づく入出力要求か否かを判別する。具体的には、制御部２１０は、入出力要求がフォーマット処理プログラム２１５の実行により生成された場合はフォーマット処理と判断し、それ以外、即ちホストプロセッサ１２から受信した場合はフォーマット処理でないと判断する（ステップ２００３）。

【００６５】

その後、制御部２１０は、仮想ボリューム１００に割当てる実領域１３２のサイズを他のデータ書き込み処理の際に割当てられる実領域１３２のサイズの１／３２と指定する。この除数はシステム管理者が任意に定めることができる。さらに、制御部２１０はステップ２００６と同様に実領域管理テーブル２２２から未使用の実領域１３２を検索する。その際、指定した大きさの空き実領域１３２がない場合は、それより大きな実領域１３２を２個に分割し、指定した大きさの空き実領域１３２を使用する。さらに制御部２１０は実領域１３２を検索した後、その実領域１３２を割当て、アクセス変換テーブル２２４、実領域管理テーブル２２２及び仮想ボリューム管理テーブル２２１を更新する。

【００６６】

図６は、第三の実施形態における、フォーマット処理の概要を示す図である。

【００６７】

図６は仮想ボリューム１００のメタデータの配置及び実領域１３２との対応を示しており、図６（１）は実領域１３２のサイズが固定されていると仮定した場合、図６（２）は実領域１３２のサイズが可変である場合を示す。図６に示した仮想ボリューム１００の記憶領域には２個のメタデータ５０２が書き込まれており、このメタデータの書き込みのためだけに、図６（１）では、２つの実領域１３２（１３２－１、１３２－２）が割当てられている。それに対し、図６（２）では、メタデータ５０２のサイズを考慮した実領域１３２－３及び実領域１３２－４が割当てられる。

【００６８】

このように、ステップ２００４で実領域１３２のサイズを縮小することで、割当てた実領域１３２中で実際には使用されていない領域を小さくすることができる。

【００６９】

次に、本発明の第四の実施形態を説明する。

【００７０】

本実施形態では、第二の実施形態と同様に、制御部２１０は、一度仮想ボリューム１００に割当てられた実領域１３２のうち、解放可能な実領域１３２を検索し解放する。ただし、実領域の解放のために使用される情報が、本実施形態ではジャーナルファイルシステムがホストプロセッサ１２で使用される場合のログ情報である点が第二の実施形態とは異なる。

【００７１】

本実施形態では、第一の実施形態の構成に加え、制御部２１０のメモリにジャーナル解析プログラム２１６が格納され、仮想ボリューム管理テーブル２２１に、ジャーナル領域のアドレス範囲を登録するエントリ３１８及びメタデータの複製を登録するエントリ３１９とが含まれる。

【００７２】

本実施形態においては、制御部２１０はジャーナル解析プログラム２１６を実行して、仮想ボリューム１００に配置されたジャーナル領域に書き込まれた情報を解析し、解放可能な実領域１３２がある場合に実領域１３２の解放処理を行う

【００７３】

ジャーナル領域のアドレス範囲が登録されるエントリ３１８は、入出力要求がジャーナル領域への書き込みか否かを判定するために制御部２１０によって参照される。仮想ボリューム１００がジャーナルファイルシステムとしてフォーマットされていない場合、ジャーナル領域のアドレス範囲エントリ３１８は空のままになる。仮想ボリューム１００がジャーナルファイルシステムとして初期化されるときには、管理者が管理コンソール１４を介して、またはジャーナルファイルシステムのフォーマットプログラムが該ジャーナルファイルシステム中のジャーナルが格納されているアドレス範囲をエントリ３１８に書き込む。メタデータの複製エントリ３１８は、制御部２１０が、仮想ボリューム１００に格納されるメタデータの複製を保存する際に使用される。

【００７４】

図７は、第四の実施形態において、入出力要求を受け付けた制御部２１０の処理手順を示すフローチャートである。

【００７５】

本処理では、第一の実施形態の図３に示した処理に加え、制御部２１０が、入出力要求がジャーナル領域への書き込みか否かを判断する処理（ステップ２００７）及び解放可能な実領域１３２の解放を行う処理（ステップ２００８）が行われる。ステップ２００７において制御部２１０は、書き込み先のアドレスがジャーナル領域範囲３１８の中にある場合に、ジャーナル領域への書き込みと判断し、ジャーナル解析プログラム２１６の処理、即ちステップ２００８へ進む。ステップ２００８において制御部２１０は、ジャーナル領域に書き込まれた情報を解析する。ジャーナル領域に書き込まれる情報とは具体的には、メタデータ３１９の一部とその（メタデータ３１９内の）オフセットであるので、これらを元に制御部２１０はメタデータ３１９を構築する。このメタデータ３１９にはファイルシステム上のファイルとそれが占めるデータ領域のアドレスとの対応が格納されているので、このメタデータ３１９の変更内容により、制御部２１０は実領域１３２の割当が不要になった領域の仮想ボリューム上のアドレスを検索する。その

後、そのアドレスを元に制御部２１０は仮想ボリューム管理テーブル２２１を検索し、見つかった実領域１３２を解放し、仮想ボリューム管理テーブル２２１、アクセス変換テーブル２２４及び実領域管理テーブル２２２を更新する。

【００７６】

図８は、第四の実施形態における、実領域１３２の解放処理の概要を示す図である。

【００７７】

図８は、仮想ボリューム１００におけるジャーナル領域、メタデータ、ファイルの配置及び実領域１３２との対応を示している。図８（１）は、ジャーナルファイルシステム上のファイル５０１の縮小前を、図８（２）は縮小後を示している。このようにファイル５０１が縮小される場合、縮小されるファイル５０１を管理しているメタデータ５０２をホストプロセッサ１２が更新するが、それに先立ち、処理内容をジャーナルログとして記録するために、ホストプロセッサ１２はジャーナル領域のエントリ５０３に情報を書き込む。このジャーナル領域のエントリ５０３には、メタデータ５０２の内容が含まれており、この内容からファイル５０１が縮小され、図８（２）に示した領域を占有するようになることが分る。

【００７８】

その結果、図８（２）の場合は、実領域１３２－４に対応する領域が空きになるため、実領域１３２－４は解放可能になる。なお、メタデータ５０２には全てのファイル５０１が仮想ボリューム１００のどの記憶領域を占有しているかを示す情報が登録されているため、制御部２１０は、メタデータ５０２に登録された内容から、実領域１３２に対応する仮想ボリューム１００の記憶領域が空きになったかどうかを判断する。さらに、制御部２１０は、ジャーナル領域のエントリ５０３へデータが書き込まれると、エントリ５０３から構築した最新のメタデータをエントリ３１９に保存し、その結果から仮想ボリューム管理テーブル２２１のエントリ３１９を更新して、実領域１３２－４を解放する。つまり、以上に述べた、ファイル５０１の縮小、ジャーナル領域のエントリ５０３の書き込み、エントリ３１９の更新、実領域１３２－４の解放はこの順で行われることになる。

【００７９】

尚、ホストプロセッサ１２によるメタデータ５０２の書き込みは、ジャーナル領域のエントリ５０３の書き込み以降に、制御部２１０の処理と非同期に行われる。そのために本実施形態ではメタデータ５０２を読み込まず、代わりにエントリ３１９を使用する。

【００８０】

さらに、ファイル５０１の削除によって実領域１３２が解放可能になる場合も、ジャーナル領域にその削除についての内容が記録されるので、上述した処理と同様に実領域１３２を解放することができる。

【００８１】

このように、制御部２１０は、仮想ボリューム１００でデータが割当てられなくなった実領域１３２を解放できる。

【００８２】

次に、本発明の第五の実施形態を説明する。

【００８３】

本実施形態では、上述した第二、第三の実施形態において仮想化スイッチ１１で実行されていたデフラグ処理プログラム２１４等の処理が専用の計算機にて行われる。

【００８４】

本実施形態では、図１で示した第一の実施形態の構成に、仮想化スイッチ１１に接続される計算機である専用サーバ１５が追加される。専用サーバ１５は、制御部２１０に代わり、デフラグ処理プログラム２１４およびフォーマット処理プログラム２１５の実行を行う。制御部２１０の処理能力やメモリ容量には限りがあり、仮想ボリューム１００の数や対応するファイルシステムの種類には限界がある。そのため、専用サーバ１５でデフラグ処理やフォーマット処理を代行する。

【００８５】

したがって、第五の実施形態においては、仮想化スイッチ１１の制御部２１０のメモリには、上述した実施形態で説明したデフラグ処理プログラム２１４やフ

ォーマット処理プログラム２１５は格納されておらず、その代わり、仮想化スイッチ１１に接続される専用サーバ１５と仮想化スイッチ１１との間の遣り取りを制御部２１０が制御するためのプログラムであるサーバ連携プログラム２１７がメモリに格納されている。

【００８６】

制御部２１０は、デフラグ処理を行うときに、デフラグ処理プログラム２１４を実行する代わりに、サーバ連携プログラム２１７を実行し、専用サーバ１５にデフラグ処理開始要求を送信し、専用サーバ１５から仮想ボリューム１００上で空きになった記憶領域のリストを受信する。次に制御部２１０は、この空きになった記憶領域のリストから対応する実領域１３２を仮想ボリューム管理テーブル２２１から検索し、見つかった実領域１３２を解放する。

【００８７】

制御部２１０は、フォーマット処理を行うときに、フォーマット処理プログラム２１５を実行する代わりにサーバ連携プログラム２１７を実行し、専用サーバ１５にフォーマット処理開始要求を送信する。その後制御部２１０の処理は第三の実施形態と同様である。

【００８８】

このように、本実施形態では、仮想ボリューム１００の数や対応するファイルシステムの種類を増やすことが出来る。

【００８９】

尚、上述の実施形態では、仮想化を実現する装置としてスイッチを例にあげて説明したが、それ以外の装置、例えば計算機でもルーターでも良い。

【００９０】

【発明の効果】

本発明によれば、効率的に計算機に記憶領域を割当てることができる。

【図面の簡単な説明】

【図１】

本発明を適用したシステムの全体構成図である。

【図２】

仮想化スイッチ１１の内部構成を示す図である。

【図３】

制御部２１０の入出力要求の処理手順を示すフローチャートである。

【図４】

第二の実施形態におけるデータの再配置の処理の概要を示す図である。

【図５】

第三の実施形態における制御部２１０の入出力要求の処理手順を示すフローチャートである。

【図６】

第三の実施形態におけるフォーマット処理の概要を示す図である。

【図７】

第四の実施形態における制御部２１０の入出力要求の処理手順を示すフローチャートである。

【図８】

第四の実施形態における、実領域１３２の解放処理の概要を示す図である。

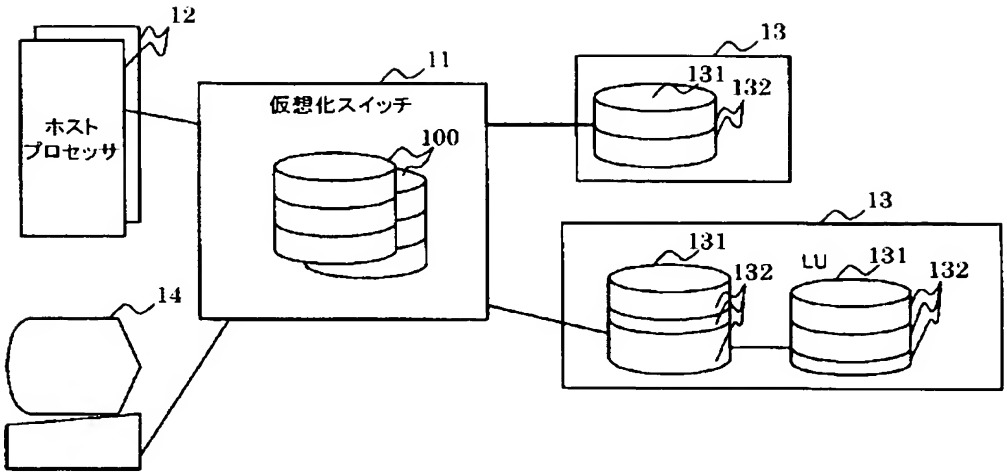
【符号の説明】

１１…仮想化スイッチ、１２…ホスト、１３…ストレージ装置、２１０…制御部、２２０…記憶部。

【書類名】　図面

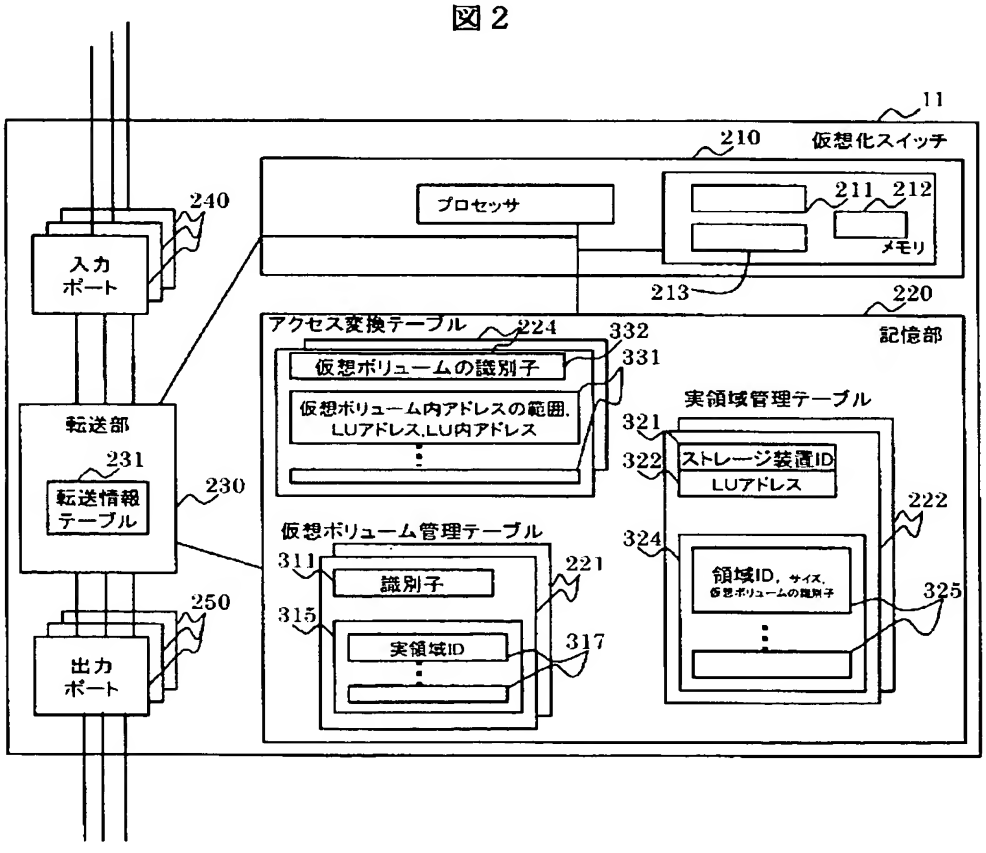
【図１】

図１



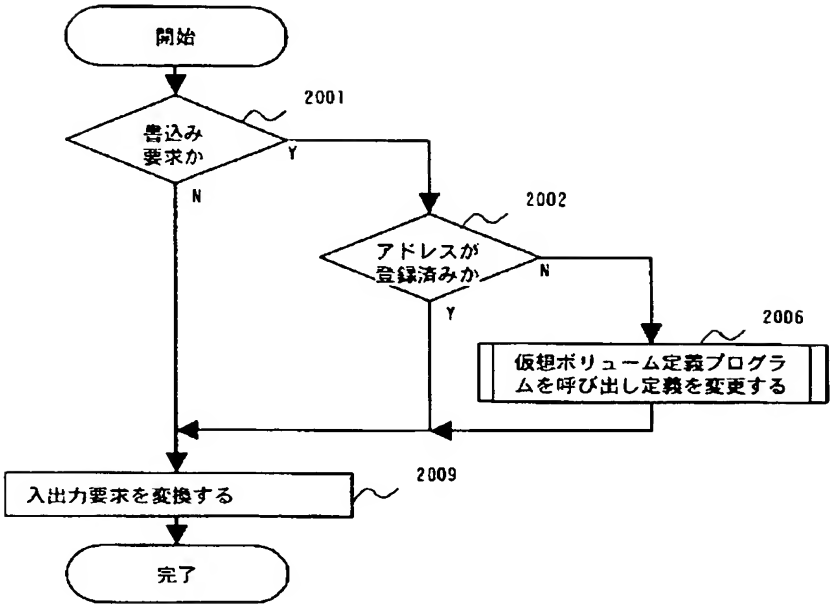
【図２】

図２



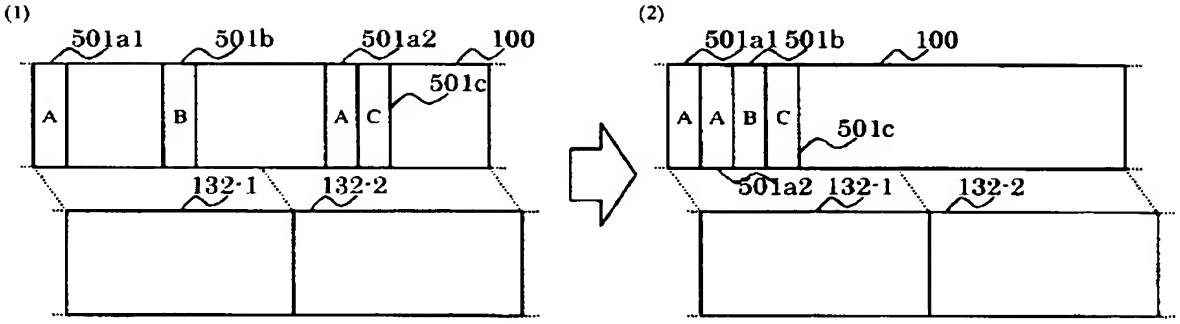
【図３】

図３


開始
2001
書込み
要求か
Y
N
2002
アドレスが
登録済みか
N
Y
2006
仮想ボリューム定義プログラムを呼び出し定義を変更する
入出力要求を変換する
2009
完了

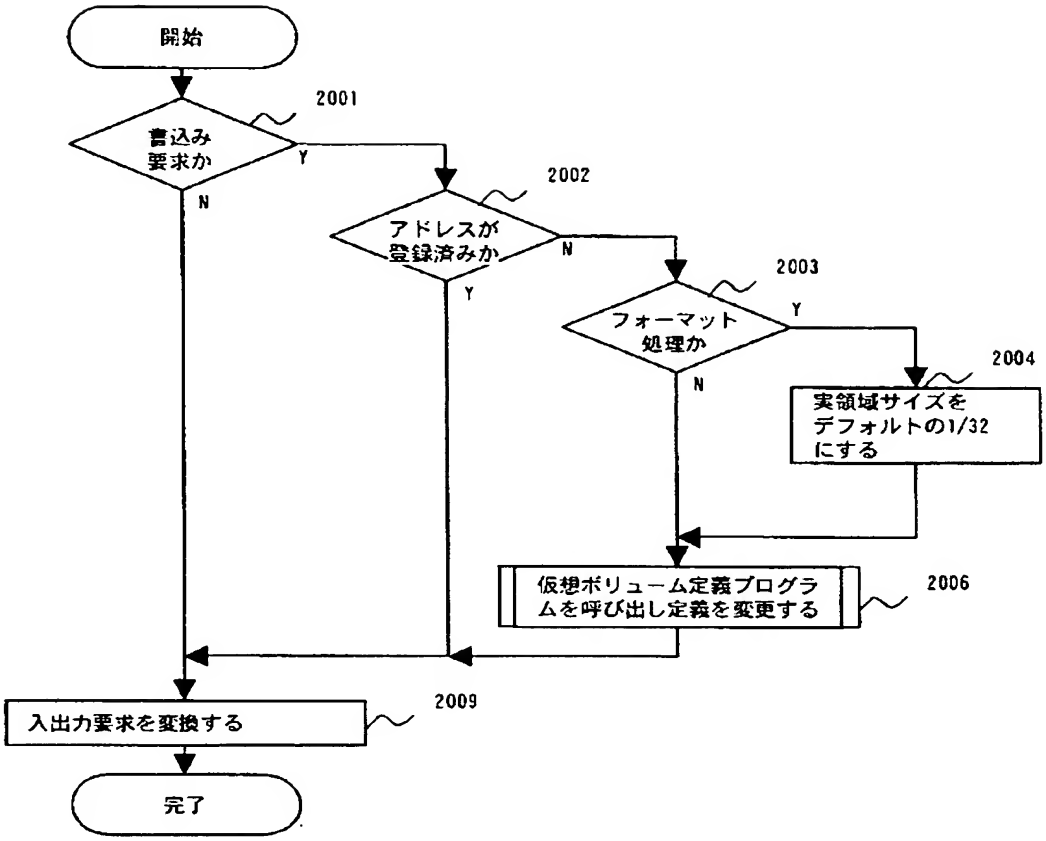
【図４】

図４


(1)
501a1
501b
501a2
100
501c
A
B
A
C
132-1
132-2
(2)
501a1
501b
100
A
A
B
C
501c
501a2
132-1
132-2

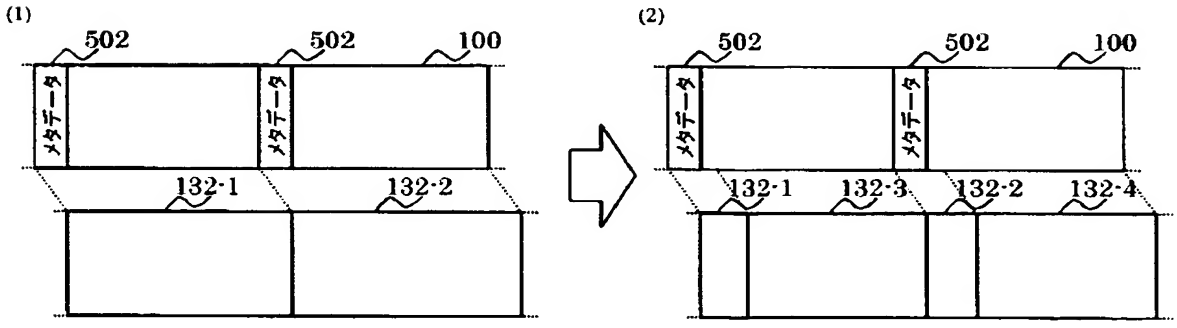
【図５】

図５


開始
2001
書込み要求か
Y
N
2002
アドレスが登録済みか
N
Y
2003
フォーマット処理か
Y
N
2004
実領域サイズをデフォルトの1/32にする
仮想ボリューム定義プログラムを呼び出し定義を変更する
2006
入出力要求を変換する
2009
完了

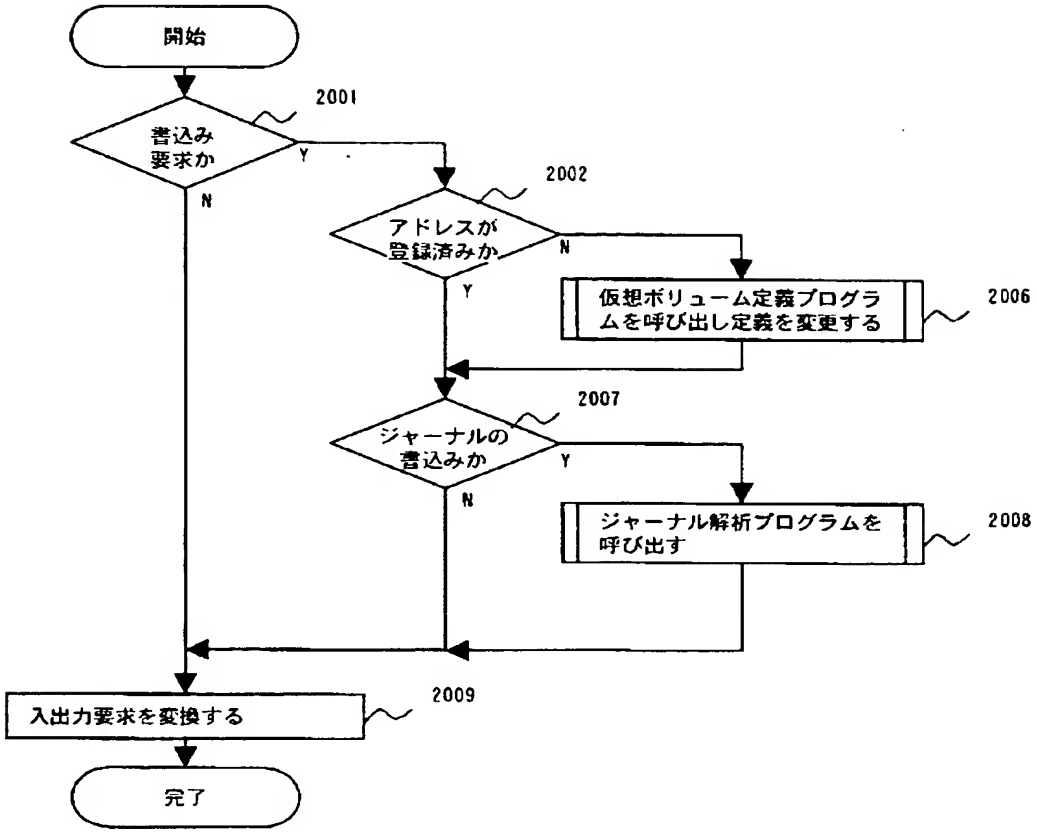
【図６】

図６


(1)
502
502
100
メタデータ
メタデータ
132-1
132-2
(2)
502
502
100
メタデータ
メタデータ
132-1
132-3
132-2
132-4

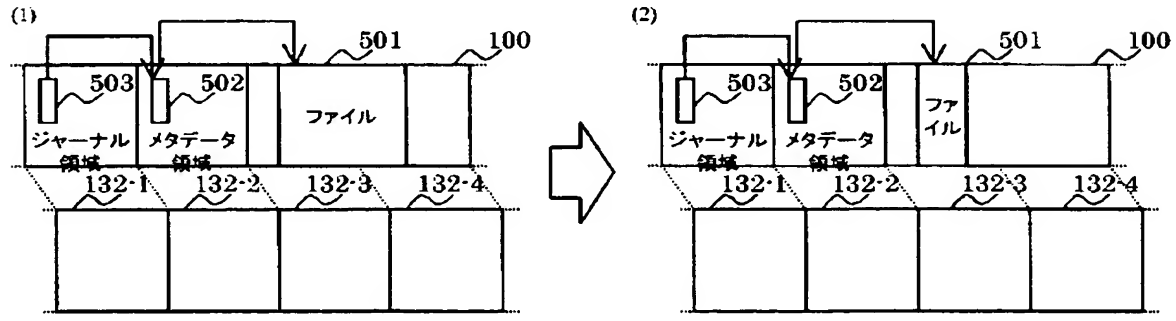
【図７】

図７


開始
2001
書込み要求か
Y
N
2002
アドレスが登録済みか
N
Y
仮想ボリューム定義プログラムを呼び出し定義を変更する
2006
2007
ジャーナルの書込みか
Y
N
ジャーナル解析プログラムを呼び出す
2008
入出力要求を変換する
2009
完了

【図８】

図８


(1)
501
100
503
502
ファイル
ジャーナル領域
メタデータ領域
132-1
132-2
132-3
132-4
(2)
501
100
503
502
ファイル
ジャーナル領域
メタデータ領域
132-1
132-2
132-3
132-4

【書類名】　要約書

【要約】

【課題】

計算機に仮想化された記憶装置システムの記憶領域を割当てる際に、効率的に割当てることが出来ない。

【解決手段】

仮想化スイッチ１１は、仮想ボリュームの作成時に、仮想ボリュームに実領域を割当てず、後にホストプロセッサ１２が仮想ボリュームへ書き込みを行ったときに初めて、書き込みが行われた仮想ボリューム上の領域に実領域を割当てる。さらに、仮想化スイッチ１１は、割当が不要となった実領域を解放する。また仮想化スイッチ１１は、入出力要求の内容に応じて、割当てる実領域の割当単位を小さくする。

【選択図】　図２

認定・付加情報

特許出願の番号	特願２００３－１５００８２
受付番号	５０３００８８０３８６
書類名	特許願
担当官	第七担当上席　　００９６
作成日	平成１５年　５月２９日

＜認定情報・付加情報＞

【提出日】　平成15年 5月28日

次頁無

特願２００３－１５００８２

出　願　人　履　歴　情　報

識別番号　　［０００００５１０８］

１．変更年月日　　１９９０年　８月３１日
［変更理由］　　新規登録
住　所　　東京都千代田区神田駿河台４丁目６番地
氏　名　　株式会社日立製作所